Robotics: Science and Systems 2026
Sydney, Australia, July 13–17, 2026

Semantically Structured Mixture-of-Experts for Compositional Robotic Manipulation

Chengyu Deng[†*], Guanqi Chen[†‡*], Yizhou Chen[†], Zejia Liu[†], Zhiwen Ruan[‡], Guanhua Chen[‡], Jia Pan[†]
[†]The University of Hong Kong [‡]Southern University of Science and Technology
[*]Equal contribution

***Abstract*—Diffusion-based policies have established a new standard for precise robotic manipulation but face a critical scalability bottleneck: high-performance models are computationally expensive, while lightweight alternatives often fail to generalize across diverse multi-task environments. Mixture-of-Experts (MoE) architectures offer a promising path to efficiency by activating only a subset of parameters. However, existing MoE routing mechanisms typically rely on low-level noise or latent statistics, ignoring the compositional nature of manipulation tasks. This can fragment reusable behaviors across experts, limiting interpretability and transferability. We introduce Semantically Structured Mixture-of-Experts Diffusion Policy (SMoDP) for compositional robotic manipulation, a framework that grounds expert specialization in semantic task structure. SMoDP leverages a lightweight, inference-time skill predictor, supervised by offline annotations from Vision-Language Models (VLMs), to route action chunks to experts specialized for specific behavioral phases. To ensure robust assignment, we propose a dual contrastive alignment strategy that grounds multi-modal observations in language-defined skill semantics (Inter-modal) while enforcing routing consistency across visually distinct but functionally related behaviors (Intra-modal). Our approach outperforms representative diffusion and MoE-based baselines on multi-task benchmarks with significantly improved parameter efficiency and demonstrates effective compositional transfer to novel tasks through parameter-efficient fine-tuning.**

I. Introduction

Diffusion-based policies have recently emerged as a powerful paradigm for robotic manipulation [6, 28, 39, 40, 4], achieving remarkable success in single-task scenarios through their ability to model multimodal action distributions and handle complex, high-dimensional control problems. By modeling action generation as conditional denoising, diffusion policies can represent multimodal action distributions and produce temporally coherent action chunks.

However, scaling these generative policies to general-purpose, multi-task settings presents a critical trilemma: we simultaneously desire (1) the precise control of diffusion policies, (2) the broad generalization of large-scale foundation models, and (3) the inference efficiency required for real-time deployment. Current approaches typically compromise on at least one front. Standard diffusion policies often struggle to generalize across diverse tasks without massive parameter scaling. Conversely, recent "robot foundation models" such as RDT [23] and Pi-series [3, 11] achieve breadth by scaling to hundreds of millions or billions of parameters. While effective, these massive architectures incur substantial computational overhead, making real-time deployment on resource-constrained robotic platforms impractical.

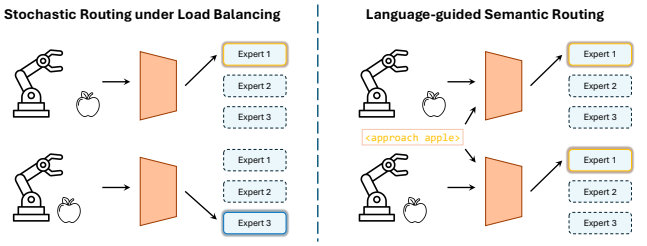

Fig. 1. Routing comparison between stochastic load-balanced routing (left) and our proposed language-guided, skill-aware semantic routing (right). Under a load-balancing objective, stochastic routing may assign different experts to generate similar or identical actions, leading to suboptimal expert assignments. In contrast, our language-guided semantic routing mechanism enables skill-aware expert assignment, where similar or identical skills are processed by the same experts.

A principled solution to this efficiency-capacity trade-off is conditional computation, specifically through Mixture-of-Experts (MoE) architectures [30, 10, 5, 22, 36]. By activating only a sparse subset of parameters per inference step, MoEs can scale model capacity without a corresponding increase in compute. Recent work such as MoDE [30] has successfully integrated MoE into diffusion policies by routing computation based on diffusion noise levels. However, these methods primarily rely on low-level signals (e.g., noise or latent variance), crucially overlooking the semantic and temporal structure of manipulation tasks. Because noise-based routing lacks explicit temporal awareness, it may fail to provide consistent guidance at the transitions between distinct behavioral phases. This leads to "routing confusion" at skill boundaries, where shared behaviors—such as "grasping"—are fragmented across disparate experts (see Figure 1), limiting both transferability and interpretability.

Recent work [13, 26] has shown that introducing structured guidance into MoE routing can promote more coherent expert specialization and improve both performance and efficiency. These approaches demonstrate that routing decisions need not be driven purely by unconstrained correlations in the training data, but can benefit from additional structural signals. However, prior methods [20, 37, 25, 33, 8] either use language or skill abstractions as an external interface for planning and skill orchestration, or discover reusable skills from latent or action-space structure, rather than explicitly using language-grounded skill semantics to organize expert routing. In contrast, we

explore skill-conditioned routing in the context of multi-task imitation learning, where skills exhibit consistent semantic meaning across tasks and directly guide expert specialization inside the policy.

In this work, we aim to transfer knowledge from existing skills to semantically similar new skills, while minimizing the number of parameters and computational cost. To this end, we propose Semantically Structured Mixture-of-Experts Diffusion Policy (SMoDP), a novel framework that integrates skill-conditioned expert routing into multi-task diffusion policies. Our approach introduces two key innovations. First, we design a VLM-aided skill abstraction module to partition a long demonstration into a sequence of semantically meaningful skills. A skill predictor learns to predict the upcoming skill from a multimodal context, enabling skill-aware routing decisions. Second, we introduce a dual contrastive learning strategy to ensure effective skill-based expert specialization: inter-modal contrastive learning aligns the skill predictor's outputs with VLM-generated descriptions, bridging states with language-based skill semantics; intra-modal contrastive learning supervises the router to produce consistent routing distributions for similar skills, encouraging functionally related skills to activate overlapping expert subsets. This dual learning paradigm enables interpretable and parameter-efficient expert specialization by ensuring that the router leverages rich semantic information while maintaining routing consistency across similar skills. By aligning the policy's modular architecture with the semantic structure of the task, SMoDP achieves the highest performance among all evaluated methods on multi-task benchmarks. It not only improves parameter efficiency but also enables compositional transfer: by fine-tuning only the skill predictor and router while freezing expert weights, the model can tackle novel tasks by recombining previously learned skill modules.

In summary, our contributions are as follows:

1) We introduce SMoDP, a semantically structured, skill-conditioned routing framework for diffusion policies that leverages language-based skill representations to guide expert selection, enabling more interpretable and structured expert specialization for multi-task robotic manipulation.
2) We develop an offline VLM-aided skill abstraction module that automatically annotates demonstrations with open-vocabulary verb–noun skill segments, providing semantic supervision without manual labels or inference-time VLM calls.
3) We propose a dual contrastive learning strategy that aligns skill prediction with language semantics and enforces routing consistency across related skills, encouraging skill-aligned expert specialization.
4) We empirically validate SMoDP on both simulation and real-world benchmarks, demonstrating improved data efficiency, parameter efficiency, and compositional transfer through parameter-efficient fine-tuning.

II. Related Work

A. Diffusion Models for Robotic Manipulation

Diffusion models [34, 9, 15], originally developed for generative modeling in vision and language, have recently been adapted for learning robot policies from demonstrations. In imitation learning, Diffusion Policy [6] formulates action generation as a conditional denoising process, enabling multimodal and temporally coherent action sequences and achieving strong performance in single-task manipulation. Subsequent work has explored more efficient or expressive variants, including flow-matching policies [41] and consistency policies [28], extensions to 3D spatial representations [40, 16], improved temporal modeling with dynamics constraints [24], and applications to long-horizon or multi-stage manipulation [4, 39, 38].

Despite these advances, generalizing diffusion policies to diverse multi-task settings remains a key challenge in robotics. One line of work addresses this challenge by scaling up model capacity and training on massive, heterogeneous datasets [23, 3, 2, 11, 27, 17]. While effective, such approaches rely on extensive data collection and large models, limiting practicality for many robotic platforms. As a result, an alternative line of research focuses on improving generalization and efficiency under data-limited conditions. Some methods learn structured latent spaces or skill abstractions to capture shared structure across tasks [20, 37, 25]. Closely related, STEER [33] explores language-based skill orchestration with a VLM in the control loop, while SMP [8] abstracts manipulation through action-space primitives. Other approaches introduce architectural innovations, such as Mixture-of-Experts (MoE) diffusion policies [30, 36], which reduce computation through sparse expert activation and enable partial parameter reuse across tasks.

However, existing approaches for multi-task diffusion policies largely rely on latent representations learned from data to capture shared structure across tasks. While effective in practice, such latent abstractions do not explicitly encode the human understanding of manipulation tasks, making it difficult to ensure consistent reuse of shared behaviors across tasks—particularly under limited data or novel task compositions. This motivates the incorporation of explicit structural inductive biases that reflect the compositional nature of manipulation skills.

B. Mixture-of-Experts

Mixture-of-Experts (MoE) [12, 14] architectures enable efficient scaling by sparsely activating subsets of parameters conditioned on the input. MoE has seen renewed interest in large language models [32, 7, 18], where sparse activation allows models with extremely large capacity to be trained and deployed efficiently. Recently, MoE architectures [10, 5, 22, 36, 30] have been adopted for robotic manipulation to address the computational and data efficiency challenges of multi-task policy learning. For example, Sparse-DP [36] explores task-specific expert specialization, while MoDE [30] integrates MoE into diffusion policies via noise-conditioned routing, significantly reducing inference cost while maintaining competitive multi-task performance.

Despite these advances, existing MoE-based imitation policies predominantly rely on data-driven routing mechanisms [7, 18, 31], where expert assignments are induced implicitly from training dynamics rather than being aligned with semantically meaningful behavioral structure. Recent work on structured routing has shown that introducing additional constraints can improve expert coherence and efficiency: CR-MoE [13] leverages consistency regularization to stabilize routing across similar inputs, and Omi et al. [26] propose similarity-preserving objectives that reduce expert fragmentation and improve load balancing. However, these approaches derive structure from latent similarity rather than from task- or skill-level semantics.

In contrast, we introduce skill-conditioned routing tailored to robotic manipulation scenarios, explicitly conditioning expert selection on language-based skill semantics. By grounding routing decisions in reusable manipulation skills, our approach promotes consistent expert reuse across tasks and task compositions, enabling more interpretable and structured expert specialization in multi-task imitation learning.

III. Method

We present *Semantically Structured Mixture-of-Experts Diffusion Policy* (SMoDP), a skill-conditioned routing framework for multi-task diffusion policies. Existing MoE routing mechanisms [30, 36] primarily rely on statistical patterns without exploiting task structure or skill composition, leading to suboptimal expert specialization where similar skills across different tasks are processed by different experts.

SMoDP overcomes this limitation through two key innovations: (i) **offline VLM-based skill abstraction** that automatically annotates demonstrations with open-vocabulary verb–noun skills, avoiding manual skill labeling; (ii) a **skill-conditioned diffusion MoE policy** that employs a lightweight online skill predictor to infer skill embeddings from multimodal context and performs chunk-consistent expert routing with dual semantic contrastive alignment, ensuring functionally related skills activate overlapping expert subsets while enabling skill-aware control. The overview of SMoDP is shown in Figure 2.

We consider multi-task imitation learning over a set of tasks $\mathcal{T} = \{\mathcal{T}_1, \ldots, \mathcal{T}_M\}$ specified by language instructions g (e.g., "Pick up the red apple"). Given demonstrations $\mathcal{D}$, our goal is to learn a single visuomotor policy $\pi_\theta(\bar{\mathbf{a}}_{n,t} \mid \mathbf{o}_{n,t}, g_n)$ that maps the current observation $\mathbf{o}_{n,t}$ and instruction g_n to an H-step action chunk $\bar{\mathbf{a}}_{n,t} = [\mathbf{a}_{n,t}, \ldots, \mathbf{a}_{n,t+H-1}]$. We train by maximizing the demonstration log-likelihood:

$$\mathcal{L}_{\mathrm{IL}} = \mathbb{E}_{(\tau_n, g_n)\sim\mathcal{D}}\, \mathbb{E}_t \Big[\log \pi_\theta(\bar{\mathbf{a}}_{n,t} \mid \mathbf{o}_{n,t}, g_n) \Big]. \tag{1}$$

A. Offline Semantic Skill Abstraction

A key challenge in implementing skill-conditioned routing is the lack of fine-grained skill annotations in existing robotic datasets. Manual labeling is prohibitively expensive and requires domain expertise. We address this by leveraging Vision-Language Models (VLMs) to automatically construct offline skill annotations from unstructured demonstrations, exploiting their zero-shot reasoning and visual understanding capabilities. These annotations are used only during training to supervise skill prediction and routing regularization; the VLM is not invoked during policy inference.

For a trajectory τ_n with task instruction g_n, we use a VLM to decompose τ_n into a sequence of K_n semantic skill segments $\mathcal{S}_n = \{s_{n,1}, \ldots, s_{n,K_n}\}$ with temporal boundaries $\{[t^{(j)}_{\mathrm{start}}, t^{(j)}_{\mathrm{end}}]\}_{j=1}^{K_n}$. Each skill is represented as a verb–noun pair (e.g., $s_{n,j} = \langle \texttt{pick up}, \texttt{red cup} \rangle$), yielding open-vocabulary skill supervision without a predefined label set. Our annotation pipeline consists of three stages:

- **Video Downsampling.** Raw observation sequences in robotic datasets are typically high-frequency, containing redundant frames that contribute little to high-level reasoning while increasing the VLM token budget. We convert the image sequence in τ_n into a video format $\mathcal{V}_n$ using a temporal downsampling factor λ: $\mathcal{V}_n = \mathrm{Downsample}(\tau_n, \lambda)$. This operation mitigates potential hallucinations by filtering out non-essential temporal jitter while preserving critical state transitions needed for accurate action recognition.
- **Semantic Skill Extraction.** The downsampled video $\mathcal{V}_n$ and task instruction g_n are jointly fed into the VLM. We employ a structured prompting strategy that enforces a canonical output format, tasking the VLM with analyzing the trajectory and distilling the behavior into a sequence of atomic skills $\mathcal{S}_n$. Crucially, we constrain the output space to *verb–noun* pairs (e.g., $s_{n,j} = \langle \texttt{pick up}, \texttt{red cup} \rangle$), which serves as a structured semantic bottleneck that enhances interpretability and compositionality.
- **Temporal Partitioning.** For each identified skill $s_{n,j} \in \mathcal{S}_n$, the VLM is prompted to estimate the corresponding temporal boundary $[t^{(j)}_{\mathrm{start}}, t^{(j)}_{\mathrm{end}}]$ within the original trajectory τ_n.

Through this process, we transform $\mathcal{D}$ into a semantically labeled dataset $\mathcal{D}_{\mathrm{sem}}$, where each timestep t is assigned a skill label $y_{n,t} = s_{n,j}$ if $t \in [t^{(j)}_{\mathrm{start}}, t^{(j)}_{\mathrm{end}}]$. These labels supervise the online skill predictor and provide semantic structure for MoE routing. The detailed prompts and examples are provided in the supplementary material.

B. Skill-Conditioned Diffusion MoE Policy

Building upon the diffusion policy formulation [6], SMoDP integrates skill semantics into expert selection through two key components: a lightweight skill predictor that efficiently infers skill phases from multimodal context at inference time, and a chunk-consistent MoE architecture that uses the predicted skill embedding to guide expert routing.

a) Diffusion Policy Over Action Chunks: Let $\mathbf{x}_0 = \bar{\mathbf{a}}_{n,t}$ denote the demonstrated action chunk. Following the standard diffusion formulation, we perturb x_0 with Gaussian noise at diffusion step $s \in \{1, \ldots, S\}$:

$$\mathbf{x}_s = \alpha_s \mathbf{x}_0 + \sigma_s \boldsymbol{\epsilon}, \quad \boldsymbol{\epsilon} \sim \mathcal{N}(\mathbf{0}, \mathbf{I}), \tag{2}$$

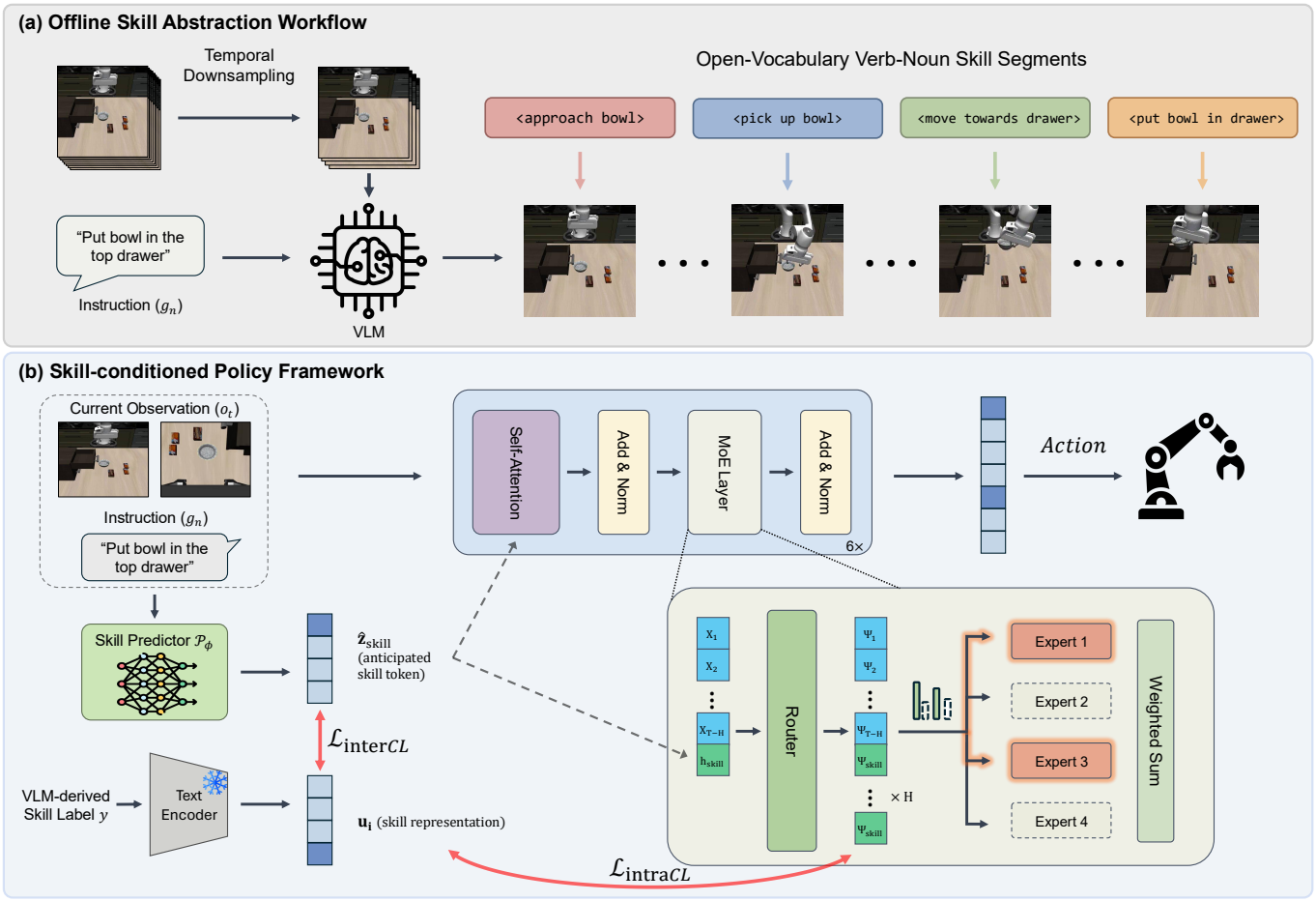

Fig. 2. Overview of SMoDP: (a) **Offline Skill Abstraction**: A workflow that automatically annotates demonstrations with open-vocabulary verb–noun skills, eliminating the need for manual labeling. (b) **Skill-Conditioned Diffusion MoE Policy**: A framework that leverages a lightweight skill predictor to anticipate the upcoming skill from multimodal context, then performs chunk-consistent expert routing with dual semantic contrastive alignment—ensuring that functionally related skills activate overlapping experts while enabling skill-aware control.

where $\{\alpha_s, \sigma_s\}$ follow a predefined noise schedule. We learn a conditional denoiser f_θ that predicts $\boldsymbol{\epsilon}$ from the noisy action chunk $\mathbf{x}_s$, observation $\mathbf{o}_{n,t}$, task instruction g_n, and diffusion step s using the denoising objective:

$$\mathcal{L}_{\text{diff}} = \mathbb{E}_{\mathbf{x}_0,s,\boldsymbol{\epsilon}}\left[\|\boldsymbol{\epsilon} - f_\theta(\mathbf{x}_s, \mathbf{o}_{n,t}, g_n, s)\|_2^2\right]. \tag{3}$$

b) Lightweight Skill Predictor: While VLM-generated skill annotations provide rich supervision during training, invoking a large VLM at inference time is impractical for real-time robotic control. We therefore train a lightweight skill predictor $\mathcal{P}_\phi$ that infers a compact semantic skill embedding from the policy's multimodal context.

Let the context token sequence be $\mathbf{C} = [\mathbf{e}_{\text{obs}}; \mathbf{e}_{\text{goal}}] \in \mathbb{R}^{T_{\text{ctx}} \times d}$, where $\mathbf{e}_{\text{obs}}$ denotes visual tokens from the observation and $\mathbf{e}_{\text{goal}}$ denotes language tokens from the task instruction. We introduce a learnable query $\mathbf{q}_{\text{skill}} \in \mathbb{R}^d$ to extract skill-relevant information via cross-attention, followed by a feed-forward projection into the frozen text-embedding space $\mathbb{R}^{d_u}$:

$$\hat{\mathbf{z}}_{\text{skill}} = \text{FFN}\Big(\text{CrossAttn}(\mathbf{q}_{\text{skill}}, \mathbf{C}, \mathbf{C})\Big). \tag{4}$$

By predicting a continuous embedding in the same space as the frozen text encoder, the skill predictor can be supervised by language-derived skill labels. This query-based design is computationally lightweight, employing only a single cross-attention layer and a compact projection module, while producing a semantic skill representation used for subsequent routing.

c) Skill-Conditioned MoE Denoising: We implement the denoiser f_θ as a Diffusion Transformer in which the feed-forward networks are replaced by MoE blocks. Let $\mathbf{H}^0 \in \mathbb{R}^{T \times d}$ denote the token sequence obtained by concatenating context tokens with H action tokens representing the noisy action chunk $\mathbf{x}_s$, where $T = T_{\text{ctx}} + H$. We condition the transformer on the predicted skill embedding through a projected conditioning vector $\mathbf{c} = \mathbf{W}_c \hat{\mathbf{z}}_{\text{skill}} \in \mathbb{R}^d$, which is broadcast to all tokens. At layer l, the update follows:

$$\begin{aligned} \mathbf{X}^l &= \mathbf{H}^{l-1} + \text{Attn}^l\Big(\mathbf{H}^{l-1} + \mathbf{1}\mathbf{c}^\top\Big), \\ \mathbf{H}^l &= \mathbf{X}^l + \text{MoE}^l\Big(\text{LN}(\mathbf{X}^l);\, \hat{\mathbf{z}}_{\text{skill}}\Big), \end{aligned} \tag{5}$$

where $\mathbf{1} \in \mathbb{R}^{T\times 1}$ denotes an all-ones vector for broadcasting, $\mathrm{LN}(\cdot)$ denotes layer normalization, and $\mathrm{Attn}^l(\cdot)$ denotes the self-attention module.

d) Chunk-Consistent Skill-Based Routing: The MoE layer incorporates the predicted skill embedding $\hat{\mathbf{z}}_{\text{skill}}$ into expert routing to encourage chunk-level consistency. Since each denoising step predicts a short action chunk, allowing different action tokens within the same chunk to select unrelated experts may introduce inconsistent local control. We therefore route all action tokens in the chunk using the same skill-conditioned router token, while keeping context-token routing token-specific.

Given token features $\mathbf{X}^l \in \mathbb{R}^{T\times d}$ at layer l, where the last H tokens correspond to the action chunk, we compute a skill token $\mathbf{h}_{\text{skill}} = \mathbf{W}_z\hat{\mathbf{z}}_{\text{skill}} \in \mathbb{R}^d$ and form the router input by replacing the action-token segment with this single skill token:

$$\tilde{\mathbf{H}}^l = \left[\mathbf{X}^l_{1:T-H},\ \mathbf{h}_{\text{skill}}\right] \in \mathbb{R}^{(T-H+1)\times d}. \tag{6}$$

Then we employ the standard multi-layer perceptron (MLP) as the router to predict the routing logits for each token:

$$\tilde{\mathbf{\Psi}}^l = \mathrm{MLP}(\tilde{\mathbf{H}}^l) \in \mathbb{R}^{(T-H+1)\times E}, \tag{7}$$

where E is the number of experts. To obtain chunk-level consistency, we broadcast the routing logits of the skill token to all H action tokens. Concretely, we define full-sequence logits $\mathbf{\Psi}^l \in \mathbb{R}^{T\times E}$ by:

$$\mathbf{\Psi}^l_{1:T-H} \leftarrow \tilde{\mathbf{\Psi}}^l_{1:T-H}, \quad \mathbf{\Psi}^l_{T-H+1:T} \leftarrow \tilde{\mathbf{\Psi}}^l_{T-H+1}. \tag{8}$$

We apply temperature-scaled softmax followed by top-k sparsification to obtain expert gates:

$$\mathbf{G}^l = \text{top-}k\Big(\mathrm{softmax}\Big(\mathbf{\Psi}^l/\tau_r\Big)\Big), \tag{9}$$

where τ_r is the router temperature and $\mathbf{G}^l$ keeps only the top-k experts for each token. Given expert networks $\{E^l_e(\cdot)\}_{e=1}^E$, the MoE output for token t is computed as a gated mixture:

$$\mathrm{MoE}^l(\mathbf{h}_t) = \sum_{e=1}^{E} G^l_{t,e}\, E^l_e(\mathbf{h}_t). \tag{10}$$

This *chunk-consistent, skill-based routing* reduces within-chunk expert switching and encourages skill-aligned expert specialization across tasks.

C. Dual Semantic Contrastive Alignment

To encourage the skill predictor and MoE router to leverage the semantic structure provided by VLM annotations, we introduce a *dual semantic contrastive alignment* strategy. This approach consists of two complementary objectives: (i) *Inter-Modal Contrastive Learning (InterCL)* that aligns the predicted skill embeddings with frozen language-encoder embeddings, bridging states with language-based skill semantics, and (ii) *Intra-Modal Contrastive Learning (IntraCL)* that supervises the router to produce consistent routing distributions for semantically similar skills, encouraging functionally related skills to activate overlapping expert subsets. InterCL shapes the predicted skill representation, whereas IntraCL directly regularizes the router behavior; thus, the former provides language-grounded skill semantics, while the latter encourages such semantics to be reflected in expert selection. Together with the diffusion denoising loss $\mathcal{L}_{\text{diff}}$ defined in Eq. 3 and the standard MoE load-balancing regularizer $\mathcal{L}_{\text{lb}}$ [7], this dual alignment encourages interpretable and parameter-efficient expert specialization.

a) Language-Derived Continuous Weights: Inspired by CWCL [35], we also adopt continuous similarity weights for skill alignment. For a minibatch of size B, let $\mathbf{u}_i = \mathrm{norm}(\mathrm{TextEnc}(y_i)) \in \mathbb{R}^{d_u}$ denote the normalized text embedding of the VLM-generated skill label y_i from a frozen text encoder. We construct a non-negative semantic affinity matrix $\mathbf{W} \in \mathbb{R}^{B\times B}$ from pairwise text-embedding similarity:

$$d_{ij} = \max\left(\langle \mathbf{u}_i, \mathbf{u}_j\rangle, 0\right), \quad d_{ii} = 0. \tag{11}$$

To reduce noise from spurious positive pairs, we retain only the top-m most similar samples for each anchor i, where $m = \lceil \rho(B-1)\rceil$ and $\rho \in (0,1]$ denotes a hyperparameter controlling the keep ratio. Let $\text{top-}m(d_i)$ denote the index set of the m largest entries in row $d_i = \{d_{ij}\}_{j=1}^B$. We define the weight matrix entries as:

$$w_{ij} = \begin{cases} d_{ij}, & j \in \text{top-}m(d_i), \\ 0, & \text{otherwise.} \end{cases} \tag{12}$$

b) Inter-Modal Contrastive Learning (InterCL): Let $\mathbf{q}_i = \mathrm{norm}(\hat{\mathbf{z}}_{\text{skill},i}) \in \mathbb{R}^{d_u}$ be the normalized predicted skill embedding for sample i. InterCL bridges the gap between the learned skill predictor and the language-based semantic space by aligning $\mathbf{q}_i$ to frozen text embeddings via a weighted contrastive loss. We compute inter-modal similarity logits:

$$\ell_{ij} = \langle \mathbf{q}_i, \mathbf{u}_j\rangle/\tau_c, \tag{13}$$

where τ_c is a temperature parameter. The InterCL loss is then formulated as:

$$\mathcal{L}_{\text{interCL}} = -\frac{1}{B}\sum_{i=1}^{B}\frac{1}{\sum_{j=1}^{B} w_{ij}}\sum_{j=1}^{B} w_{ij}\,\log\frac{\exp(\ell_{ij})}{\sum_{k=1}^{B}\exp(\ell_{ik})}. \tag{14}$$

Since $\mathbf{u}$ is produced by a pre-trained language encoder, this alignment encourages $\hat{\mathbf{z}}_{\text{skill}}$ to inherit compositional structure and generalize by semantic proximity rather than memorizing a closed set of skill IDs.

c) Intra-Modal Contrastive Learning (IntraCL): While InterCL aligns predicted skill embeddings with language semantics, it does not directly constrain the router behavior. To encourage that semantically similar skills induce similar routing patterns, we introduce IntraCL, which operates on the routing logits themselves.

Let $\mathbf{\Psi}^l_i \in \mathbb{R}^{T\times E}$ be the broadcasted routing logits at MoE layer l for sample i, and let $\mathcal{L} \subseteq \{1,\dots,L\}$ denote the set of MoE layers where we apply routing alignment. We flatten and normalize the routing logits into a router feature vector:

$$\mathbf{r}^l_i = \mathrm{norm}\Big(\mathrm{vec}(\mathbf{\Psi}^l_i)\Big) \in \mathbb{R}^{TE}. \tag{15}$$

We then compute intra-modal similarity logits:

$$\ell_{ij}^{l} = \langle \mathbf{r}_i^l, \mathbf{r}_j^l \rangle / \tau_c. \tag{16}$$

The IntraCL loss encourages routing consistency across semantically similar skills:

$$\mathcal{L}_{\text{intraCL}} = \frac{1}{|\mathcal{L}|} \sum_{l \in \mathcal{L}} \left[-\frac{1}{B} \sum_{i=1}^{B} \frac{\sum_{j=1}^{B} w_{ij} \log p_{ij}^{l}}{\sum_{j=1}^{B} w_{ij}} \right], \tag{17}$$

$$\text{where} \quad p_{ij}^{l} = \frac{\exp(\ell_{ij}^{l})}{\sum_{k=1}^{B} \exp(\ell_{ik}^{l})}.$$

By leveraging the language-derived weights w_{ij}, IntraCL acts as a topology-preserving constraint that compels the router to respect semantic distances: skills that are close in language space are encouraged to activate similar expert subsets. This regularization enhances expert specialization by preventing arbitrary partitioning across functionally related behaviors.

d) Overall Training Objective: We combine the diffusion denoising loss, the standard MoE load-balancing regularizer, and the two semantic alignment terms:

$$\mathcal{L} = \mathcal{L}_{\text{diff}} + \lambda_{\text{lb}} \mathcal{L}_{\text{lb}} + \lambda_{\text{inter}} \mathcal{L}_{\text{interCL}} + \lambda_{\text{intra}} \mathcal{L}_{\text{intraCL}}, \tag{18}$$

where λ_{lb}, λ_{inter}, and λ_{intra} balance the load-balancing, InterCL, and IntraCL losses, respectively. This objective trains the model to denoise actions effectively while aligning both its skill representations and routing behavior with meaningful semantic structure derived from language.

IV. Experiments

In this section, we evaluate SMoDP in both simulated and real-world robotic manipulation settings to assess whether skill-conditioned expert routing improves multi-task learning, data efficiency, compositional transfer, and real-world execution. Using the LIBERO benchmark and a real bimanual ALOHA system, we compare SMoDP against representative diffusion, MoE-based, and skill-structured imitation learning baselines, together with targeted ablations. Our experiments are designed to validate four core claims: (i) skill-aware routing improves multi-task performance, (ii) language-grounded skill structure improves data utilization in low-data regimes, (iii) frozen experts can be reused for parameter-efficient few-shot compositional transfer, and (iv) semantic routing improves multi-task learning ability on a real robot.

A. Experimental Setup

a) Simulation Benchmark: We evaluate our method on the LIBERO suite [21], a standard multi-task manipulation benchmark with diverse object interactions and long-horizon compositions. We use the following task suites: (i) LIBERO-90, containing 90 manipulation tasks; (ii) LIBERO-10, consisting of 10 long-horizon tasks derived from LIBERO-90; (iii) LIBERO-OBJECT and LIBERO-GOAL, designed to stress object-centric and goal-conditioned generalization, respectively; and (iv) LIBERO-GOAL-OOD [19], a skill-level split constructed from LIBERO-OBJECT/GOAL to

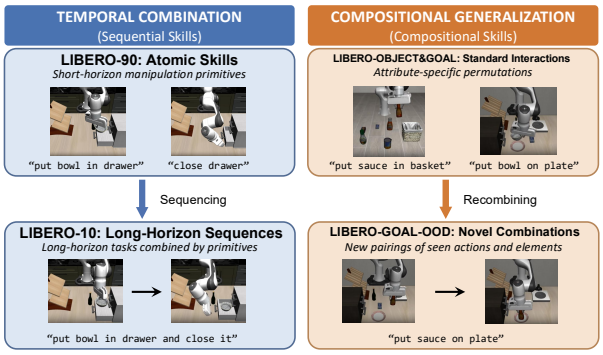

Fig. 3. Overview of tasks in 4 LIBERO simulation task suites.

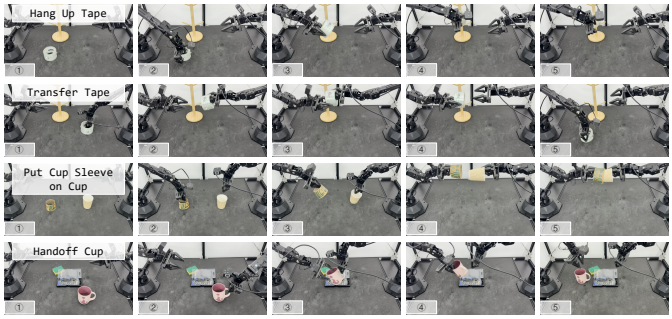

Fig. 4. Real-world dual-arm ALOHA setup and task illustrations.

test compositional generalization to unseen skill combinations (e.g., "put bowl on stove" + "put bottle on cabinet" ⇒ "put bottle on stove"). For task suites LIBERO-90/10/OBJECT/GOAL, 50 demonstrations are provided per task. For LIBERO-GOAL-OOD, we manually collect 10 demonstrations per task and sub-sample them for few-shot adaptation (1/5/10 demos). Figure 3 shows some examples of these 4 task suites. Each episode contains two RGB views, including a fixed front camera and a wrist-mounted camera. All methods are trained and evaluated using the same observations, demonstrations, and rollout protocol.

b) Real-world Benchmark: We additionally validate our approach on a real ALOHA system for bimanual multi-task manipulation. Illustrations and real-world task setups are shown in Figure 4. Our setup uses a 7-DoF master–puppet teleoperation system with four RGB cameras, including two wrist-mounted cameras and two external cameras from the main front view and the top-down view. Unlike the single-arm simulation benchmark, the real-world ALOHA tasks require bimanual coordination and frequent hand–object occlusions; therefore, we provide the policy with a short visual history consisting of the most recent three image frames from all four cameras. We collect 20 demonstrations per task for four real-world training tasks: (i) *Hang Up Tape*—pick up a tape roll with the left arm and hang it onto a shelf hook; (ii) *Transfer Tape*—pick up a tape roll with the right arm, hand it over to the left arm, and place it on the table; (iii) *Put Cup Sleeve on Cup*—pick up both cup and cup sleeve, and put cup sleeve on

cup; and (iv) *Handoff Cup*—pick up cup and hand it off to the other hand.

c) Baselines: We compare SMoDP against representative imitation learning baselines, including diffusion-policy baselines with transformer or CNN backbones (DP-T, DP-CNN) [6], a vector-quantized transformer policy (QUEST) [25], and MoE-based diffusion policies. Specifically, we include Sparse Diffusion Policy (SDP) [36], which performs task-conditioned sparse expert activation, and Mixture-of-Denoising Experts (MoDE) [30], a transformer-based diffusion policy that routes tokens to sparse denoising experts conditioned on the diffusion noise level. MoDE (Pretrain) denotes initializing MoDE from large-scale OXE dataset [27] pre-training checkpoint before fine-tuning on the target LIBERO suite. For transfer experiments, we additionally include: (i) MoDE with LoRA fine-tuning after pre-training on the source suite (MoDE+LoRA), and (ii) MoDE trained from scratch on the target suite (MoDE-scratch).

d) Implementation Details: For offline VLM skill annotation, we use Qwen3-VL [1] with a temporal downsampling factor of $\lambda = 5$. For the diffusion transformer, we use 6 transformer layers, 4 experts per MoE layer, and top-2 expert activation. In simulation, we condition on the most recent observation frame from the two RGB views; in real-world ALOHA experiments, we use the most recent three frames from the four RGB cameras. In both settings, the policy predicts an action chunk with horizon $H = 10$. For positive sample selection in our dual contrastive learning, we set the keep ratio to $\rho = 0.1$. We use Sentence-BERT [29] as the frozen text encoder to extract embeddings of VLM-generated skill annotations for dual semantic contrastive alignment. VLM annotation and contrastive alignment are used only during training; at inference time, SMoDP uses the learned lightweight skill predictor and does not call the VLM. In our implementation, SMoDP takes approximately 4 hours to train on a single NVIDIA RTX 4090. With 10 denoising steps, the average inference time is about 121 ms per action chunk, which is comparable to standard MoE diffusion policies.

B. Multi-task Performance on LIBERO-10 and LIBERO-90

We first evaluate multi-task imitation learning on LIBERO-10 and LIBERO-90 using 50 demonstrations per task. The results are shown in Figure 5. Overall, SMoDP achieves the highest average success rate on both benchmarks among the evaluated methods. Notably, SMoDP also outperforms MoDE (Pretrain), which spends more than 3 days pre-training on the large-scale OXE dataset. This suggests that language-grounded skill routing provides an effective inductive bias for sharing manipulation behaviors across tasks, allowing the policy to learn strong multi-task visuomotor representations without relying on external large-scale pre-training data. Compared with the fairer non-pretrained MoE baselines, the improvement indicates that the gain is not merely due to sparse expert capacity, but also comes from organizing expert activation around reusable skill semantics.

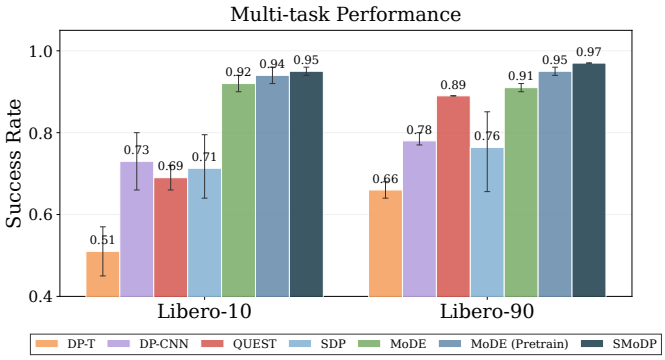

Fig. 5. Average results for LIBERO-10 and LIBERO-90 averaged over 3 seeds with 20 rollouts per task.

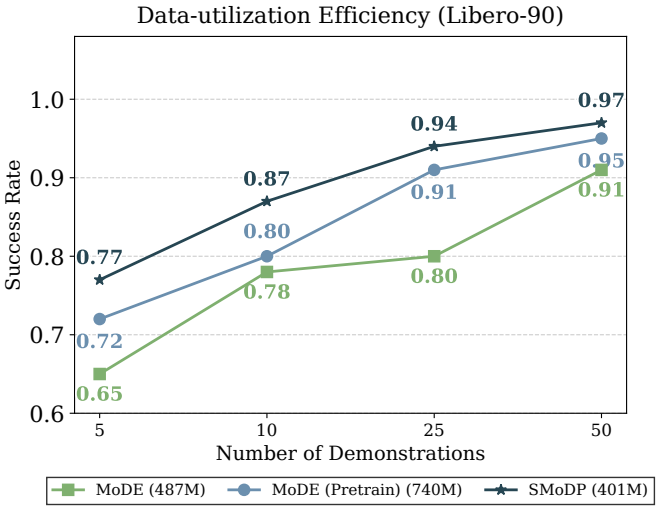

Fig. 6. Average success rate on Libero-90 under different numbers of demonstrations for training.

C. Data Utilization Efficiency

We evaluate data utilization efficiency by training on only 10%/20%/50% of LIBERO-90 demonstrations, corresponding to 5/10/25 demonstrations per task. This setting tests whether language-derived skill structure helps the policy reuse shared behaviors when task-specific demonstrations are scarce. Figure 6 summarizes model performances under different numbers of demonstrations. SMoDP consistently outperforms MoDE and MoDE (Pretrain) across all data regimes, while using fewer total parameters. Notably, SMoDP trained with 50% of the demonstrations achieves performance comparable to MoDE (Pretrain) trained with the full target dataset, despite MoDE (Pretrain) using additional OXE pre-training and about $1.85\times$ more parameters. With a comparable parameter count, SMoDP also shows better data efficiency than MoDE: using only 50% of the training demonstrations, SMoDP outperforms MoDE trained on the full dataset. These results indicate that semantic skill supervision and skill-aware routing are particularly beneficial in low-data regimes, where purely data-driven routing is more likely to overfit task-specific correlations.

TABLE I
FEW-SHOT TRANSFER SUCCESS RATE (↑) ON LIBERO-10 AND LIBERO-GOAL-OOD.

Method	Parameters (M)		LIBERO-10			LIBERO-GOAL-OOD		
	Trainable	Total	1 Demo	5 Demos	10 Demos	1 Demo	5 Demos	10 Demos
MoDE+LoRA	**13.6**	487	0.245	0.425	0.509	0.319	0.410	0.635
MoDE-scratch	487	487	0.085	0.245	0.519	0.170	0.129	0.495
SMoDP (Ours)	13.7	**401**	**0.520**	**0.765**	**0.840**	**0.320**	**0.590**	**0.660**

TABLE II
REAL-WORLD ROBOT EXPERIMENT RESULTS ON FOUR TASKS

Methods	Success Rate (Success / Total)				Avg. Success Rate
	Hang Up Tape	Transfer Tape	Put Cup Sleeve on Cup	Handoff Cup	
MoDE	15/20 (75%)	11/20 (55%)	12/20 (60%)	0/20 (0%)	47.50%
SMoDP (Ours)	**19/20 (95%)**	**18/20 (90%)**	**16/20 (80%)**	**20/20 (100%)**	**91.25%**

D. Task Transfer and Compositional Generalization

We evaluate whether SMoDP supports compositional transfer by reusing frozen experts and adapting only the skill predictor and router under few-shot target demonstrations. This setting tests whether previously learned experts can serve as reusable skill modules, while the lightweight skill predictor and router learn to recombine them for new task compositions.

a) LIBERO-90 → LIBERO-10*:* We first pre-train the model on LIBERO-90 with 50 demos/task and then fine-tune it on LIBERO-10 using 1/5/10 demos per task. During fine-tuning, SMoDP freezes all expert parameters and updates only the skill predictor and router. We compare against two baselines: MoDE+LoRA (fine-tuning attention weights with a matched trainable-parameter budget) and MoDE-scratch (training from scratch on LIBERO-10). Table I shows that SMoDP achieves the best performance across all three few-shot transfer settings, outperforming MoDE+LoRA by an average margin of 31.5 percentage points on LIBERO-10. Moreover, SMoDP updates only 13.7M parameters, accounting for 3.4% of the full model. These results suggest that the source-suite experts capture reusable manipulation behaviors, and that new long-horizon tasks can be learned by adapting the skill prediction and routing modules rather than retraining the full policy.

b) LIBERO-OBJECT/GOAL → LIBERO-GOAL-OOD *:* LIBERO-GOAL-OOD is constructed by recombining skill segments from LIBERO-OBJECT and LIBERO-GOAL, creating unseen verb–noun compositions at test time. We fine-tune the model with 1/5/10 demonstrations per OOD task. We use the same adaptation protocol as above: SMoDP updates only the skill predictor and router, while baselines employ MoDE+LoRA or MoDE-scratch. The results are shown in Table I. SMoDP matches the strongest baseline in the 1-demo setting and achieves clear improvements in the 5-demo and 10-demo settings. Unlike noise-conditioned routing in MoDE, SMoDP explicitly aligns routing decisions with language-grounded skill semantics, making adaptation focus on recombining reusable behaviors rather than relearning the entire action distribution from few demonstrations. This allows the skill predictor and router to adjust expert selection for new skill compositions, enabling parameter-efficient transfer to OOD task combinations.

E. Real-world Results

We further evaluate SMoDP in real-world multi-task learning settings on four bimanual manipulation tasks. Table II reports the success rate over 20 rollouts per task. All methods are trained from scratch using the same demonstrations. SMoDP consistently outperforms MoDE across all four tasks, improving the average success rate from 47.50% to 91.25%. On *Hang Up Tape*, both methods achieve relatively high success rates, while SMoDP further improves performance from 75% to 95%. On *Transfer Tape*, which involves picking up the tape, handing it over, and placing it on the table, SMoDP improves the success rate from 55% to 90%. On *Put Cup Sleeve on Cup*, SMoDP improves from 60% to 80%, indicating more stable execution in object engagement and alignment. The largest gap appears on *Handoff Cup*, where MoDE fails in all trials whereas SMoDP succeeds in all 20 rollouts.

We attribute these improvements to SMoDP's semantic skill decomposition and skill-aware routing, which enable knowledge sharing across related phases while preserving stable expert specialization. Qualitative inspection further suggests that baseline failures often occur around high-variance transition points, such as grasping, handover, and object engagement. By explicitly modeling latent skills and routing actions through semantically structured experts, SMoDP better disambiguates these critical phases and maintains stable execution in real-world multi-task settings.

F. Ablation Studies

We validate the proposed framework by ablating its key components and architectural choices. Specifically, we study: (i) InterCL, which aligns predicted skill embeddings with language semantics; (ii) IntraCL, which encourages semantically

TABLE III
ABLATION STUDIES FOR SMODP ON LIBERO-90.

Variant	Avg. Success Rate ± Std
SMoDP (full)	**0.970 ± 0.002**
w/o InterCL	0.958 ± 0.002
w/o IntraCL	0.957 ± 0.008
w/o InterCL & IntraCL	0.946 ± 0.011
experts/layer = 3	0.957 ± 0.005
experts/layer = 6	0.958 ± 0.005
experts/layer = 8	0.950 ± 0.004

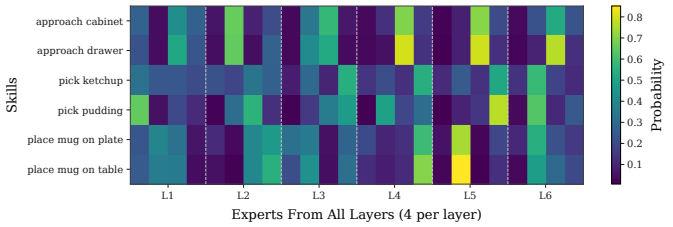

Fig. 7. Expert activation heatmap for semantic skills across MoE layers. For each skill, we average expert activation probabilities over all occurrences of that skill during LIBERO-90 evaluation rollouts. Columns correspond to experts from layers 1–6, with 4 experts per layer.

consistent routing distributions; and (iii) the number of experts per MoE layer. The ablation results are presented in Table III.

Removing either InterCL or IntraCL leads to a performance drop from 0.970 to 0.958 and 0.957, respectively, while removing both further decreases the success rate to 0.946. This indicates that the two objectives are complementary: InterCL improves language-grounded skill representation, whereas IntraCL directly regularizes routing behavior.

For the number of experts, using 4 experts per layer achieves the best performance among the tested settings. Reducing the number of experts to 3 may limit specialization capacity, while increasing it to 6 or 8 does not further improve performance and may fragment the training signal across more experts. We therefore use 4 experts per layer as a trade-off between capacity, specialization, and parameter efficiency. With 6 MoE layers and top-2 routing, 4 experts per layer already provide sufficient routing diversity while keeping the model size comparable to MoDE.

We further visualize the learned routing behavior in Figure 7. For each semantic skill, we average the expert activation probabilities over all occurrences of that skill and display the activation pattern across all MoE layers. The heatmap shows that different skills induce distinct activation patterns, while semantically related skills exhibit similar expert usage. For example, related approach skills share more similar routing patterns than object-specific picking or placing skills. This provides qualitative evidence suggesting that SMoDP learns skill-dependent expert specialization beyond simple load balancing.

G. Discussion and Limitations

While SMoDP demonstrates significant improvements in semantic-aware multi-task learning, several limitations remain to be addressed in future work.

Coarse-grained Semantics: Encoding skills as verb-noun pairs provides a structured bottleneck but overlooks fine-grained control dynamics. For example, $\langle \texttt{pick up}, \texttt{cup} \rangle$ omits variations in speed, force, or trajectory, limiting the model's ability to adapt execution styles within the same high-level skill.

VLM-dependent Annotations: Offline annotation quality hinges on the VLM's zero-shot reasoning and priors. Coarse segmentation may miss subtle transitions, while excessive granularity can cause unstable routing. Visual challenges (e.g., motion blur, occlusion) further risk hallucinations, introducing noisy supervision via inaccurate boundaries or mislabeled skills.

Long-tail Skill Distribution: Robotic datasets often follow a long-tail distribution, where frequent primitives (e.g., reach, pick up) dominate over rare but critical skills (e.g., unscrew lid). This imbalance biases the router toward common skills, reducing expert specialization for infrequent actions.

V. CONCLUSION

We introduce SMoDP, a semantically structured, skill-conditioned routing framework for diffusion policies that leverages language-based skill representations to guide expert selection, enabling more interpretable and structured expert specialization for multi-task robotic manipulation. Experiments on both simulation and real-world benchmarks demonstrate that SMoDP achieves superior data efficiency and parameter efficiency, while enabling compositional transfer through parameter-efficient fine-tuning.

ACKNOWLEDGMENTS

This work was supported by the Natural Science Foundation of China (62461160309), the NSFC-RGC Joint Research Scheme (N_HKU705/24), and Hong Kong RGC (GRF 17201025, GRF 17200924).

REFERENCES

[1] Shuai Bai, Yuxuan Cai, Ruizhe Chen, Keqin Chen, Xionghui Chen, Zesen Cheng, Lianghao Deng, Wei Ding, Chang Gao, Chunjiang Ge, Wenbin Ge, Zhifang Guo, Qidong Huang, Jie Huang, Fei Huang, Binyuan Hui, Shutong Jiang, Zhaohai Li, Mingsheng Li, Mei Li, Kaixin Li, Zicheng Lin, Junyang Lin, Xuejing Liu, Jiawei Liu, Chenglong Liu, Yang Liu, Dayiheng Liu, Shixuan Liu, Dunjie Lu, Ruilin Luo, Chenxu Lv, Rui Men, Lingchen Meng, Xuancheng Ren, Xingzhang Ren, Sibo Song, Yuchong Sun, Jun Tang, Jianhong Tu, Jianqiang Wan, Peng Wang, Pengfei Wang, Qiuyue Wang, Yuxuan Wang, Tianbao Xie, Yiheng Xu, Haiyang Xu, Jin Xu, Zhibo Yang, Mingkun Yang, Jianxin Yang, An Yang, Bowen Yu, Fei Zhang, Hang Zhang, Xi Zhang, Bo Zheng, Humen Zhong, Jingren Zhou, Fan Zhou, Jing Zhou, Yuanzhi Zhu, and Ke Zhu. Qwen3-vl technical report, 2025. URL https://arxiv.org/abs/2511.21631.

[2] Jose Barreiros, Andrew Beaulieu, Aditya Bhat, Rick Cory, Eric Cousineau, Hongkai Dai, Ching-Hsin Fang, Kunimatsu Hashimoto, Muhammad Zubair Irshad, Masha Itkina, et al. A careful examination of large behavior models for multitask dexterous manipulation. *arXiv preprint arXiv:2507.05331*, 2025.

[3] Kevin Black, Noah Brown, Danny Driess, Adnan Esmail, Michael Equi, Chelsea Finn, Niccolo Fusai, Lachy Groom, Karol Hausman, Brian Ichter, Szymon Jakubczak, Tim Jones, Liyiming Ke, Sergey Levine, Adrian Li-Bell, Mohith Mothukuri, Suraj Nair, Karl Pertsch, Lucy Xiaoyang Shi, James Tanner, Quan Vuong, Anna Walling, Haohuan Wang, and Ury Zhilinsky. π_0: A vision-language-action flow model for general robot control, 2026. URL https://arxiv.org/abs/2410.24164.

[4] Yizhou Chen, Hang Xu, Dongjie Yu, Zeqing Zhang, Yi Ren, and Jia Pan. Svip: Sequencing bimanual visuomotor policies with object-centric motion primitives. *arXiv preprint arXiv:2506.18825*, 2025.

[5] Baiye Cheng, Tianhai Liang, Suning Huang, Maanping Shao, Feihong Zhang, Botian Xu, Zhengrong Xue, and Huazhe Xu. Moe-dp: An moe-enhanced diffusion policy for robust long-horizon robotic manipulation with skill decomposition and failure recovery. *arXiv preprint arXiv:2511.05007*, 2025.

[6] Cheng Chi, Zhenjia Xu, Siyuan Feng, Eric Cousineau, Yilun Du, Benjamin Burchfiel, Russ Tedrake, and Shuran Song. Diffusion policy: Visuomotor policy learning via action diffusion. *The International Journal of Robotics Research*, page 02783649241273668, 2023.

[7] William Fedus, Barret Zoph, and Noam Shazeer. Switch transformers: Scaling to trillion parameter models with simple and efficient sparsity. *Journal of Machine Learning Research*, 23(120):1–39, 2022.

[8] Ce Hao, Xuanran Zhai, Yaohua Liu, and Harold Soh. Abstracting robot manipulation skills via mixture-of-experts diffusion policies. *arXiv preprint arXiv:2601.21251*, 2026.

[9] Jonathan Ho, Ajay Jain, and Pieter Abbeel. Denoising diffusion probabilistic models. *Advances in neural information processing systems*, 33:6840–6851, 2020.

[10] Suning Huang, Zheyu Aqa Zhang, Tianhai Liang, Yihan Xu, Zhehao Kou, Chenhao Lu, Guowei Xu, Zhengrong Xue, and Huazhe Xu. Mentor: Mixture-of-experts network with task-oriented perturbation for visual reinforcement learning. In *International Conference on Machine Learning*, 2025.

[11] Physical Intelligence, Kevin Black, Noah Brown, James Darpinian, Karan Dhabalia, Danny Driess, Adnan Esmail, Michael Equi, Chelsea Finn, Niccolo Fusai, Manuel Y. Galliker, Dibya Ghosh, Lachy Groom, Karol Hausman, Brian Ichter, Szymon Jakubczak, Tim Jones, Liyiming Ke, Devin LeBlanc, Sergey Levine, Adrian Li-Bell, Mohith Mothukuri, Suraj Nair, Karl Pertsch, Allen Z. Ren, Lucy Xiaoyang Shi, Laura Smith, Jost Tobias Springenberg, Kyle Stachowicz, James Tanner, Quan Vuong, Homer Walke, Anna Walling, Haohuan Wang, Lili Yu, and Ury Zhilinsky. $\pi_{0.5}$: a vision-language-action model with open-world generalization, 2025. URL https://arxiv.org/abs/2504.16054.

[12] Robert A Jacobs, Michael I Jordan, Steven J Nowlan, and Geoffrey E Hinton. Adaptive mixtures of local experts. *Neural computation*, 3(1):79–87, 1991.

[13] Ziyu Jiang, Guoqing Zheng, Yu Cheng, Ahmed Hassan Awadallah, and Zhangyang Wang. Cr-moe: Consistent routed mixture-of-experts for scaling contrastive learning. *Transactions on Machine Learning Research*, 2024.

[14] Michael I Jordan and Robert A Jacobs. Hierarchical mixtures of experts and the em algorithm. *Neural computation*, 6(2):181–214, 1994.

[15] Tero Karras, Miika Aittala, Timo Aila, and Samuli Laine. Elucidating the design space of diffusion-based generative models. *Advances in neural information processing systems*, 35:26565–26577, 2022.

[16] Tsung-Wei Ke, Nikolaos Gkanatsios, and Katerina Fragkiadaki. 3d diffuser actor: Policy diffusion with 3d scene representations. In *Conference on Robot Learning*, pages 1949–1974. PMLR, 2025.

[17] Alexander Khazatsky, Karl Pertsch, Suraj Nair, Ashwin Balakrishna, Sudeep Dasari, Siddharth Karamcheti, Soroush Nasiriany, Mohan Kumar Srirama, Lawrence Yunliang Chen, Kirsty Ellis, et al. Droid: A large-scale in-the-wild robot manipulation dataset. In *Robotics: Science and Systems*, 2024.

[18] Dmitry Lepikhin, HyoukJoong Lee, Yuanzhong Xu, Dehao Chen, Orhan Firat, Yanping Huang, Maxim Krikun, Noam Shazeer, and Zhifeng Chen. Gshard: Scaling giant models with conditional computation and automatic sharding. In *International Conference on Learning Representations*, 2021.

[19] Quanyi Li. Task reconstruction and extrapolation for π_0 using text latent, 2025. URL https://arxiv.org/abs/2505.03500.

[20] Zhixuan Liang, Yao Mu, Hengbo Ma, Masayoshi Tomizuka, Mingyu Ding, and Ping Luo. Skilldiffuser: Interpretable hierarchical planning via skill abstractions in diffusion-based task execution. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 16467–16476, 2024.

[21] Bo Liu, Yifeng Zhu, Chongkai Gao, Yihao Feng, Qiang Liu, Yuke Zhu, and Peter Stone. Libero: Benchmarking knowledge transfer for lifelong robot learning. *Advances in Neural Information Processing Systems*, 36:44776–44791, 2023.

[22] Chaoqi Liu, Haonan Chen, Sigmund H. Høeg, Shaoxiong Yao, Yunzhu Li, Kris Hauser, and Yilun Du. Flexible multitask learning with factorized diffusion policy. *IEEE Robotics and Automation Letters*, 11(4):4697–4704, 2026. doi: 10.1109/LRA.2026.3664611.

[23] Songming Liu, Lingxuan Wu, Bangguo Li, Hengkai Tan, Huayu Chen, Zhengyi Wang, Ke Xu, Hang Su, and Jun Zhu. Rdt-1b: a diffusion foundation model for bimanual

manipulation. In *International Conference on Learning Representations*, 2025.

[24] Xiao Liu, Yifan Zhou, Fabian Weigend, Shubham Sonawani, Shuhei Ikemoto, and Heni Ben Amor. Diff-control: A stateful diffusion-based policy for imitation learning. In *2024 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*, pages 7453–7460. IEEE, 2024.

[25] Atharva Mete, Haotian Xue, Albert Wilcox, Yongxin Chen, and Animesh Garg. Quest: Self-supervised skill abstractions for learning continuous control. *Advances in Neural Information Processing Systems*, 37:4062–4089, 2024.

[26] Nabil Omi, Siddhartha Sen, and Ali Farhadi. Load balancing mixture of experts with similarity preserving routers, 2025. URL https://arxiv.org/abs/2506.14038.

[27] Abby O'Neill, Abdul Rehman, Abhiram Maddukuri, Abhishek Gupta, Abhishek Padalkar, Abraham Lee, Acorn Pooley, Agrim Gupta, Ajay Mandlekar, Ajinkya Jain, et al. Open x-embodiment: Robotic learning datasets and rt-x models: Open x-embodiment collaboration 0. In *2024 IEEE International Conference on Robotics and Automation (ICRA)*, pages 6892–6903. IEEE, 2024.

[28] Aaditya Prasad, Kevin Lin, Jimmy Wu, Linqi Zhou, and Jeannette Bohg. Consistency policy: Accelerated visuomotor policies via consistency distillation. In *Robotics: Science and Systems*, 2024.

[29] Nils Reimers and Iryna Gurevych. Sentence-bert: Sentence embeddings using siamese bert-networks. In *Proceedings of the 2019 Conference on Empirical Methods in Natural Language Processing*. Association for Computational Linguistics, 11 2019. URL https://arxiv.org/abs/1908.10084.

[30] Moritz Reuss, Jyothish Pari, Pulkit Agrawal, and Rudolf Lioutikov. Efficient diffusion transformer policies with mixture of expert denoisers for multitask learning. In *International Conference on Learning Representations*, 2025.

[31] Carlos Riquelme, Joan Puigcerver, Basil Mustafa, Maxim Neumann, Rodolphe Jenatton, André Susano Pinto, Daniel Keysers, and Neil Houlsby. Scaling vision with sparse mixture of experts. *Advances in Neural Information Processing Systems*, 34:8583–8595, 2021.

[32] Noam Shazeer, Azalia Mirhoseini, Krzysztof Maziarz, Andy Davis, Quoc Le, Geoffrey Hinton, and Jeff Dean. Outrageously large neural networks: The sparsely-gated mixture-of-experts layer. In *International Conference on Learning Representations*, 2017.

[33] Laura Smith, Alex Irpan, Montserrat Gonzalez Arenas, Sean Kirmani, Dmitry Kalashnikov, Dhruv Shah, and Ted Xiao. Steer: Flexible robotic manipulation via dense language grounding. In *2025 IEEE International Conference on Robotics and Automation (ICRA)*, pages 16517–16524. IEEE, 2025.

[34] Yang Song and Stefano Ermon. Generative modeling by estimating gradients of the data distribution. *Advances in neural information processing systems*, 32, 2019.

[35] Rakshith Sharma Srinivasa, Jaejin Cho, Chouchang Yang, Yashas Malur Saidutta, Ching-Hua Lee, Yilin Shen, and Hongxia Jin. Cwcl: Cross-modal transfer with continuously weighted contrastive loss. *Advances in Neural Information Processing Systems*, 36:78496–78513, 2023.

[36] Yixiao Wang, Yifei Zhang, Mingxiao Huo, Thomas Tian, Xiang Zhang, Yichen Xie, Chenfeng Xu, Pengliang Ji, Wei Zhan, Mingyu Ding, et al. Sparse diffusion policy: A sparse, reusable, and flexible policy for robot learning. In *Conference on Robot Learning*, pages 649–665. PMLR, 2025.

[37] Kun Wu, Yichen Zhu, Jinming Li, Junjie Wen, Ning Liu, Zhiyuan Xu, and Jian Tang. Discrete policy: Learning disentangled action space for multi-task robotic manipulation. In *2025 IEEE International Conference on Robotics and Automation (ICRA)*, pages 8811–8818. IEEE, 2025.

[38] Hang Xu, Yizhou Chen, Dongjie Yu, Yi Ren, and Jia Pan. Bikc+: Bimanual hierarchical imitation with keypose-conditioned coordination-aware consistency policies. *IEEE Transactions on Automation Science and Engineering*, 23:1064–1079, 2025.

[39] Dongjie Yu, Hang Xu, Yizhou Chen, Yi Ren, and Jia Pan. Bikc: Keypose-conditioned consistency policy for bimanual robotic manipulation. *arXiv preprint arXiv:2406.10093*, 2024.

[40] Yanjie Ze, Gu Zhang, Kangning Zhang, Chenyuan Hu, Muhan Wang, and Huazhe Xu. 3d diffusion policy: Generalizable visuomotor policy learning via simple 3d representations. In *Proceedings of Robotics: Science and Systems (RSS)*, 2024.

[41] Fan Zhang and Michael Gienger. Affordance-based robot manipulation with flow matching, 2025. URL https://arxiv.org/abs/2409.01083.